ANALOGICAL REASONING IN PROBLEM SOLVINC

David T. Chen* and Nicholas V. Findler
Dept. of Computer Science, SUNY at Buffalo
Amherst, N.y. 14226

INTRODUCTION

Our ability to cope with the environment is the result of continual learninn from experience. Learnina "makes sense" onlv because situations resemble each other, and some information obtained in one situation is of use in others — or, expressed more broadly, similar situations call for similar actions. We also learn to extract the essential characteristic features of the situations we become familiar with, which then help us in retrievina the situations <u>most similar</u> to the one we are faced with.

Similarity can be of different types and at different levels of complexity. Its discovery is often the result of a goal oriented process associated with problem solving activity. Once similarity betxween a problem with a known solution and an unsolved problem has been recognized, Analogical Reasoning (AR) is normally called into action by humans to generate possible solutions to be tested then for adequacy.

We have attempted to investigate Alt detached from specific tasks and to formulate its general principles. Our objective has been to create an AR component for problem solving proarams and to assume for it the level of generality the means-ends analysis was shown to have in GPS.

WORKING HYPOTHESES OF ANALOGICAL REASONING

The underlying rationale of[1-] AR can be expressed by a few working hypotheses:

(i) Each problem is describable aS an (ordered) collection of, possibly overlapping, features. (A feature represents one or several chunked properties. Whereas properties are atomic and directly measurable, features can in general be measured as present or absent only.) The identification and extraction of features are, however, left either to a program component external to AR or to the user.

(ii) Solutions are associated with respective problems in a well-defined, deterministic manner. This assumption goes beyond the usual concept of causality. It requires that the features be identifiable and strongly enough correlated with the solutions so that the latter can be derived directly from the former.

(iii) In the task domains of interest to us, similar problems have similar solutions. (Similarity must be measured along certain dimensions that depend on a <u>priori</u> features of both problems and solutions.)

(iv) When two problems have similar solutions, the features present in one problem but not in the other are likely to be of lesser importance. In turn, features shared by problems which have similar solutions are likely to be important. These can be strengthened quantitatively the more problems, which have similar solutions, share a feature the more important it is likely to be. Also, the more features are shared by two problems the more similar their solutions will be.

THE CONTRIBUTIVE AND HIERARCHICAL MODELS

A decision must be made as to what Information should be extracted from "raw" experience with problems and solutions, and how this information should be used in determining solutions to new problems. In the <u>contributive model</u>, the features of a problem are associated with the appropriate solution steps or segments. The frequency of prior usage of a solution segment for problems with a given feature provides a heuristic auide in selecting and ordering for testing that solution segment for a new problem possessing the same feature.

The <u>hierarchical model</u> also considers the relevance of individual features. The solution segments offered for testing are selected in an order based on matching with a set of features hierarchically structured in the knowledge base. The learning process hore not only enters new information about problems, their features and sequence of solution seaments, as above, but it may also rearrange the feature hierarchy.

We feel some combination of the above two models can deal also with high order similarities — such as theorems of duality, structural, semantic, functional and thematic similarities — if sophisticated feature extraction programs can be employed in cooperation with the AR component.

PROGRA^iriED EXPLORATIONS

First, we carried out a few preliminary experiements with simple tasks. These included the contributive model for the "simulation" of some piece-wise smooth functions and the hierarchical model for several concent formation tasks.

We have then implemented a simplified system consistina of three main components: The first ornanizes and coordinates the loaic of problem solvina and is responsible for sub-problem generation. The second is the task-independent AR component offering recommendations to the previous one. Finally, a data base separate from the rest contains the task domain description, the procedures to interpret the semantics of the definitions used, and a list of all possible features. This system was used for two areas sufficiently disjoint for generality consinderations but having identical machine representation of problem constituents — theorem proving and construction tasks in plane geometry.

The objective was to prove that AR works with non-trivial problems rather than to compete with previous impressive accomplishments in the above areas. We were also able to show that the knowledge acquisition system is both effective and efficient. New concepts, such as circle, can be introduced on-line, in a high-level manner. The system is flexible, open ended and capable of accepting extensions.

Now a'E the Dept. of Comp" Science, Univ. of Missouri, Columbia, 110 65201